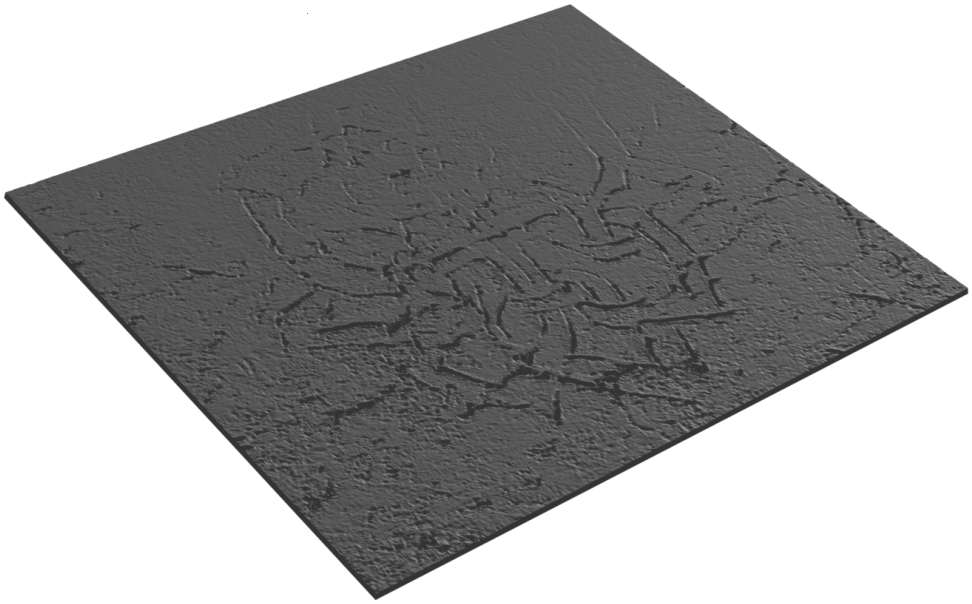


Budget efficient 3D-mesh of engravings using photometric stereo

Master's thesis in Computer science and engineering

Edvard Heinmetz

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

Master's thesis 2025

Budget efficient 3D-mesh of engravings using photometric stereo

Edvard Heinmetz

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2025

Budget efficient 3D-mesh of engravings using photometric stereo

Edvard Heinmetz

Supervisor: Ulf Assarsson, Department of Computer Science and Engineering
Advisor: Jonathan Westin, GRIDH
Examiner: Erik Sintorn, Department of Computer Science and Engineering

Master's Thesis 2025
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A mesh produced by the algorithm presented in this thesis.

Typeset in LaTeX
Gothenburg, Sweden 2025

Budget efficient 3D-mesh of engravings using photometric stereo

Edvard Heinmetz
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

This paper describes a process for calculating approximate light positions in a photo set captured of an inscribed wall, with the purpose of applying photometric stereo. The light sources across the photo set are of constant intensity and their positions are inferred using one reflective indicator sphere and the lighting of the wall. The light positions are then used to create a depth map of the wall in an optimization scheme that alternates between depth and albedo values. Various strategies for using the depth map to highlight and isolate inscriptions are then applied.

Keywords: photometric stereo, mesh, engravings, optimization.

Acknowledgements

I want to thank Jonathan Westin at Gothenburg Research Infrastructure in Digital Humanities for providing the project idea and the photos, my supervisor at Chalmers Ulf Assarsson, and my examiner Erik Sintorn.

Edvard Heinmetz, Gothenburg, 2025-06-21

Contents

List of Figures

1

Introduction

The University of Gothenburg along with other universities around the world has a long ongoing effort of digitalizing historical remains. The benefits of digitalization are multifaceted, with advantages in domains such as distribution, preservation and digital analysis.

From a distribution perspective, it removes the need for researchers to physically visit the site while performing their analysis. If accessibility is handled accordingly, a digital version of historical data could also allow universities worldwide to access the findings at any time, which widens the target group of analysts who can examine the findings.

In the aspect of preservation, digital versions of findings have the advantage of not deteriorating over time, unlike many physical historical sites and artifacts. Sites of this nature have been the focus for the University of Gothenburg recently, especially since an increasingly variable climate and new conflict zones place many previously stable historical sites at risk.

A digital version of an artifact or site also allows for digital analysis of the object. This could either be an aid for examination by a human researcher, with examples such as image manipulation, or allowing algorithmic examination where algorithms and/or AI-models can find features of interest on their own.

This project aims to process images captured in a church in Kiev, Ukraine. The photos have been taken with the intent of digital conservation of the church. Conservation has become an issue due to the church being under threat of destruction, owing to the ongoing invasion from Russia. The church as a building is in itself well documented, as it has been a frequently visited landmark for centuries, with many historical and contemporary accounts in detailed specification of renovations, architecture and its different appearances through the ages.

The project currently underway at Gothenburg Research Infrastructure in Digital Humanities (GRIDH) is aimed at a less documented and still largely uncharted aspect of the church's historical importance, namely the many inscriptions on its walls.

Figure 2 highlights how crucial lighting is for the appearance of the engravings. Can portable, non-specialty equipment produce a 3D reconstruction of a wall with inscription readability comparable to that achieved with specially manufactured

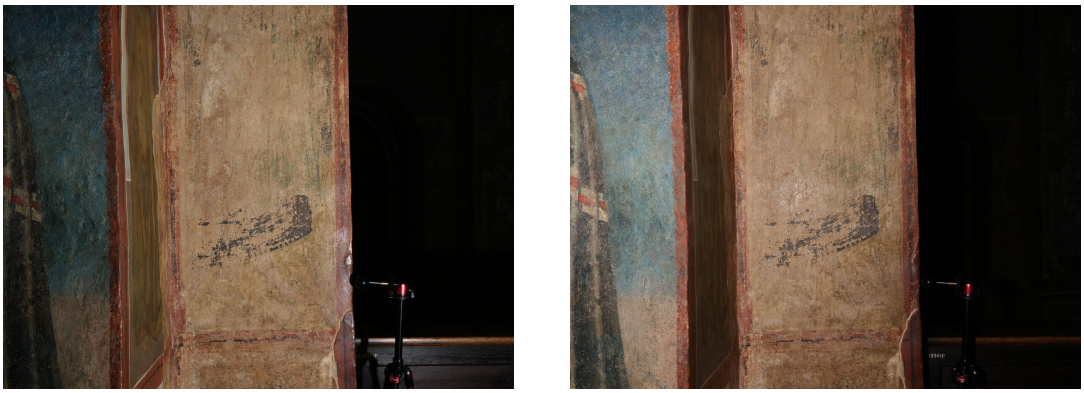


Figure 1: Two photos from the photo set, with lighting from different directions

Figure 2: A section with engravings, lit from different directions

reflectance transformation imaging equipment?

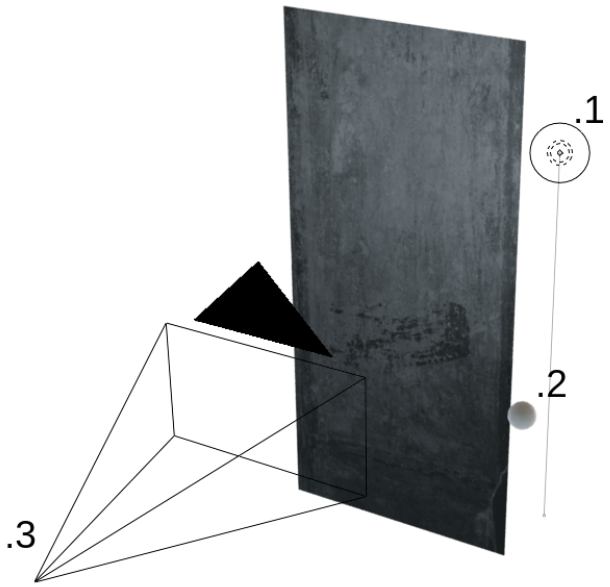
1.1 Problem

RTI (Reflectance Transformation Imaging) is a well-developed field with specialty manufactured equipment such as RTI-domes which produce scans that closely resemble the real objects. These setups require resources that are not always readily available. For one, they are expensive, but there are additional factors to consider when it comes to historical sites.

The output of RTI domes consists of a large number of data points that might be difficult to accommodate for using limited equipment. Also, the power requirements for such an advanced setup usually requires either that the pictures are taken in a grid connected facility, or with specialty batteries. Further, the setup and dismantling time for RTI-specialty equipment make it hard to digitalize a large area such as the walls in this church, while time might also become a strapped resource due to the church being in a war zone.

GRIDHs project DIGICURE[1], is aimed at preservation of endangered cultural heritage. While many of the preserved sites have had the resources and communications available for specific RTI-equipment, a part of the project is to explore the possibility

of using more budget effiecient methods to achieve similar results. Also surveying methods using equipment that can be sourced without a working postal service and on short notice, if a historical site is placed at unforseen and immediate risk. The photo set used in this project was captured with a standard DSLR camera, a LED light source, and a reflective sphere to indicate the light direction.



1. Portable LED light source, off-camera, moved with each image
2. Reflective sphere indicating light direction, mounted on a tripod
3. DSLR camera, Fujifilm GFX100 II, mounted on a tripod

Figure 3: Equipment used during capture

The sphere used in this project was specifically manufactured to be mounted on a threaded tripod. However, a reflective sphere was chosen by GRIDH because it is easy to substitute with more common objects such as a spherical doorknob or a decorative sphere coated with reflective spray paint.

1.2 Goals and Challenges

The pictures were taken with the intention of producing a normal map, an image which encodes surface directions in the RGB-channels, to digitally relight the inscriptions on the wall using an already available tool. This project aims to use the same photo set to create a 3-dimensional mesh of the wall.

A mesh is preferable over a normal map, as the self-shadowing it can produce with digital lighting is important to the appearance of engravings. The main goal is to produce a mesh that achieves multiple criteria. First and foremost that the readability of engravings while digitally examining the mesh should preferably outperform the readability in relit images. Another criterion is that the Topography Visualization Toolbox[2] developed by GRIDH should be used, to detect and highlight engravings on walls, could possibly be used to visualize and extract features from a mesh, which is not possible with a normal map.

There are multiple challenges that are known from the beginning of the project. Primarily difficulties in producing a mesh good enough to preserve small details, such as thin or shallow engravings.

These issues stem from, for example in estimating the reflectance of the wall. Since the wall is painted and engraved in different phases of its history, the reflectance of the wall is varied across its area, so reflectance has to be calculated at each pixel. When reflectance is unknown, there are multiple ways of estimating it, however. Some of them require material samples which can not be taken from the wall as the imaging, firstly, has to be non-invasive, and secondly, the team which took the pictures are no longer in Ukraine. Luckily, the photo set is quite large (50 images) when compared to the sets used by other algorithms (e.g [3][4]) which successfully estimate the reflectance, so an accurate estimate could still be within reach.

Another factor in the reflectance problem is that while light direction is known at one point in the scene, due to the indicator sphere, the distance the light shines from is unknown. Many photometric-stereo algorithms assume small objects with lights at a relatively far distance, which results in the light direction effectively being uniform across the object, known as far lighting. While our light position is fixed in one dimension, the wall is large, and the source is placed within a few meters from the wall, making the light direction non-uniform across its span. This is known as near light photometric stereo, which usually assumes the lights are either pre-calibrated in a fixed setup, or more than one sphere is placed in the scene, to fix the light sources in 3 dimensions. Since the photo set was captured to be used with another tool, which assumes far lighting, there was only one sphere used during capture. Light cues from the wall itself will be used along with the sphere to estimate the light position.

1.3 Previous Work

Photometric stereo in itself has been researched since the 70s with a pioneering paper on the subject by Woodham [5]. This paper states what defines the photometric stereo approach and what problems it is developed to solve. The algorithm is defined as developing a reflectance map for a point, then using different known light directions to uniquely determine the surface gradient as it becomes the only unknown variable in the reflectance map. The reflectance map is defined by Horn in [6], and is a function relating how much incoming light from a given angle reaches the camera.

This approach is theoretically similar to what has to be done for this project, although aspects are missing. As stated in the problems section, there are a number of uncertainty-inducing parameters such as varying reflectance, uncalibrated light sources and non-uniform light distribution. Solutions to these aspects of the problem are the subject of more recent research, and are still being explored.

Hertzmann [7] handles a similar problem, letting the object act as its own material sample while providing a specular sphere for light direction detection. In these aspects it is similar to the problem at hand, although it differs in others. The objects were small and thus assumed far lighting, while the depth map creation of the wall

has to account for near lighting. They also deliberately avoid self-shadowing of the objects while lighting them, while self shadowing will have to be accounted for in this project.

Since the image of the wall can be used to calculate light direction. Together with the spherical light indicator, analysis can be performed on the wall itself to locate the light using preprocessing of the image set, similar to what is described by McGuigan and Christmas in [8]. A method presented in the paper is accentuating the light direction over the photographed object by taking an image from the photo set, calculating its luminance, then subtracting the average luminance across the entire photo set. An example:

Figure 4: Luminance of an image from the photo set, the average luminance of the entire photo set, and their difference.

Methods of applying photometric stereo have been tested in the field of historical engravings before, such as in a paper by Solem and Nau [9]. They use photometric domes, which are half-domes placed on top of inscriptions blocking any interfering light, inside the domes there are precalibrated light sources. The results produced by these methods are of a much higher quality than the scope of this project, but as stated in the problem description, the novelty of this approach would be budget efficiency and the possibility of scanning a large area with minimal setup time and equipment. The intended quality of the scans is enough for readability and improved relighting, while the ones in Solem and Naus paper are enough to conduct research on topics such as tools used in engraving and the possibility of finding multiple works done by one artist through similar engraving technique.

For the express purpose of creating a depth map from real world noisy data with uncertain light sources, Queau et al's paper implements an optimization scheme optimizing directly for the depth map and albedo values [3]. The method presented in this paper is based around this with adaptions made to further the purpose of preserving inscriptions and minute variations in surface depth.

1.4 Approach

The process consists of calculating an expression for the expected light intensity at each pixel for each light source. Then a minimization problem is formulated by combining estimated reflectance and light directions, and each pixel's depth and reflectance estimates are refined to minimize the difference between the calculated expected light intensity and the actual level observed in the image set. Once the depth information is obtained, the resulting 3D data is saved for further analysis. The results are evaluated by assessing the readability of inscriptions when digitally relighting the produced 3D model and with digital examination methods.

1.5 Limitations

The scope of this project is to evaluate whether the photo set can be processed into a model. Thus, the calculations and programming are limited to processing this specific photo set. While a general purpose tool would be more usable in practice, the scope of creating such a tool is more suitable for a separate project. That would entail more UX to tailor to a user not familiar with photometric stereo, and versatility to generalize the calculations to other walls. Although, when methods specific to this wall are employed, a discussion is included, evaluating the assumptions made and whether these assumptions can be generalized to other walls. If a strategy is employed which is overly specific and likely would not work in other cases, the aim is to suggest other methods that could achieve the same result in a similar real world scenario, for example through additional measurements taken during photography, added reference objects, or similarly feasible solutions not requiring specialized equipment.

2

Methodology

The end goal is a three-dimensional model of the wall, and the analysis performed upon it is the subject of the report. To make the motivation of the methodology clear, this section will describe how this is achieved step by step from the raw data available. The model is created from a depth map, with one vertice represented by one pixel in the depth map. The calculation of the depth map itself relies on photometric stereo methods, which require the light position relative to each pixel making on the wall. The positioning of the light, the camera, the indication sphere, and the wall is required in relation to a chosen origin. This will be done using computer vision and mathematical methods. The methodological time plan of the project is thus these steps in reverse order.

2.1 Positioning

Positioning the wall relatively to the camera will be done using calibration techniques from computer vision. GRIDH has supplied the focal length and sensor dimensions for the camera used while capturing the photo set, these are called the camera's intrinsic properties. This reduces the amount of unknown variables in the calibration. What is left to calibrate is then the so-called extrinsic parameters of the camera, its position and rotation. The estimation of the extrinsic parameters will largely follow the process described by Zhang in [10].

The results of this calibration are measurable by comparing a digitally replicated version of the setup using the estimated coordinates of the wall and the indicator sphere. They are placed in front of a virtually defined camera using the same provided camera properties and overlaid with an actual image from the photo set, if there is a mismatch between the estimated position and the image there is a calculation error. This comparison should be done before continuing to the optimization step, as any error in the positioning will propagate as additional error sources in the later optimization.

2.2 Photometric Stereo

After the relative positioning of the wall, light, and camera, all data needed for the photometric stereo is available. The implementation of the optimization based photometric stereo is the most difficult and time-consuming step in the project, as

the mathematics involved are complex and the well researched algorithms have some discrepancies to the data this project will handle. The most important concepts and established techniques related to photometric stereo have their own descriptions below.

2.2.1 Lighting model

For this project the Lambertian lighting model is assumed[11], which assumes that incoming light is scattered equally in all directions. It is also shaded by the albedo, refered to as ρ_j for a pixel j, which is unique for every pixel. Only diffusely reflected light intensity will be considered, so ρ becomes a measure of how much incoming light a point reflects, independent from viewing angle. The formula for reflected light is further affected by the geometric term ψ_j, also unique for every pixel, a three dimensional vector encompassing light direction and distance to the light source encoded in its magnitude. This geometric term scales the incoming light into a pixel's region of the wall by the dot product with its surface normal n_j, a steeper lighting angle results in less incoming light. This project also assumes square falloff, the light intensity decreases according to the inverse square of the distance. The full formula for the lighting intensity I_j for a pixel j in the model is hence:[12]

$$I_j = \rho_j(n_j \cdot \psi_j)$$

The terms are explained in more detail in the methodology. This model has some discrepancies to the actual wall, mainly that it does not account for specular highlights or self shadowing. How these errors are accounted for is also further explained in the methodology.

2.2.2 Reflectance Modelling

In this project, each pixels reflected light to the camera from the incoming light directions is evaluated empirically from the photo set. Once the reflectance is calculated for a pixel, then its 3D-position, the camera position, and the light positions are used to determine what surface normal for the point represented in the pixel minimizes the error of the measured light in every image.

The main concept of use in this is ratio images, approximating a pixels albedo from the photo set by taking the average light intensity of a pixel through the entire photo set, then dividing every pixel in the photo set by its average value.

The motivation to do this is to isolate the light intensity from the albedo. If ρ is a good enough albedo estimate, the lambertian equation can be refactored to isolate incoming light intensity. This is used for light source estimation in a manner similar to figure 4 but using division instead of subtraction.

$$I_j = \rho_j(n \cdot \psi)$$

This conversion turns the observed value in the ratio image into:

$$\frac{I_j}{\rho_j} = n \cdot \psi$$

[13]

2.2.3 Depth map creation

The survey [12] covers multiple different approaches to creating a normal map through photometric stereo. Most algorithms described and measured in the survey are sorted into the bracket termed minimization, minimizing the error between the expected angles calculated using reflectance maps. The minimization method is a good candidate for this project because there have been methods of constraining the problem using parameters available in the data provided by GRIDH. There are a number of different constraints to be considered and possibly implemented.

Integrability is a well established constraint in shape reconstruction problems, as any real world surface is integrable, meaning it does not have any disjunctions. Since the depth map is split into pixels representing a small area, there will inevitably be some disjunction between adjacent depths, but continuation can be simulated by penalizing large differences in surface normals or high frequency depth variations.

Another enforceable constraint using our data originates in the fact the approximate shape of our object is known. It is a wall of known measurements with inscriptions, and while it cannot be assumed that the wall was built completely flat, it will differ at most a few centimeters across its span, and inscriptions are likely only a few millimeters deep, so any points outside such a span could be heavily penalized as well.

Further, the above constraints can be combined with possible fixed points, such as the corners of the wall or areas that are known to be without inscribed dents, where the depth is likely to be the same as the edge. However, this is likely not entirely true, as the wall might be convex or concave to some degree of local or global curvature. These fixed points might still constrain the normal map to a domain where the global curvature is lost, while the inscriptions remain readable, which is still within the goal.

2.2.4 Evaluation

This step is also harder to measure than the positioning, because there is no clear right or wrong depth map without measuring the physical wall with specific RTI-equipment. Although an approximate evaluation of these results is achievable by looking at the source images and comparing where the human eye can see inscriptions to the produced mesh. Evaluation is then based on if they appear like they expectedly would. Another strategy of measurement augments this same concept is artificial relighting of the normal map, and trying to position the light in a way similar to images from the set and evaluating if the accentuated inscriptions from the real image appear in the relit version as well.

2.2.5 Analysis

After all steps are finished, the final output will be analyzed in itself, by observation, relighting and texturing. Since the most sought after quality in the output is readability of the inscriptions, the report will focus on this domain. With detailed examination of the inscriptions and whether inscriptions seen in the photo set is carried over to the model to a degree that preserves readability. An analysis through GRIDHs purposely trained AI-model Topology Visualization Toolbox (TVT)[2] to emphasize inscriptions will also be applied on the 3D-model.

2.3 Positioning

Every subsequent step of the process requires the relative positioning of the camera and the light source and an initial depth estimate of the wall. The camera is first positioned using techniques from computer vision, then the camera's positioning and intrinsic operation is used to position the indicator sphere. The indicator sphere is then used in combination with the variation in brightness over each image to position the light source.

In the positioning stage of the project, there is one central parameter concerning the camera that is in focus, the projection matrix M. This is used to map three-dimensional world coordinates to two-dimensional image coordinates residing in the image plane. M can in turn be broken down into two factors, given their own explanations below.

2.3.1 The K matrix

The intrinsic matrix, denoted K, is named as such because it describes the internal operation of the camera:

$$\begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Where f is the focal distance measured in pixels for the x and y axis. c_x and c_y are the principal point coordinates, the point on the image sensor where the center of the projection lies, also measured in pixels from the top left corner of the image. γ is the skew, which is used when the axis of the image are not entirely perpendicular.

Mainly the f-values are of interest for our project. γ is simply set to 0 as the axis in our image are orthogonally aligned. The u values are also initially solved for in a simple manner, simply dividing the width and height of the image in half. This is enough for the positioning step, the principal point will however be different during the optimization, since the images will be cropped to contain only the region of interest currently being optimized. This will place the principal point outside of the current area center and the K matrix will have to be corrected.

A bit more thought has to go into the focal lengths as well, there are two distinct f variables as some cameras do not have perfectly square pixels. Thus the focal length, the distance from the focal point to the sensor, would have a different value measured in pixels along the y compared to the x axis. To determine the focal length we consult the image metadata. The data entered below are of use to our calculations.

Data	Denotation	Value
Camera		Digital Camera GFX100 II
Sensor Size x (mm)	w_s	43.8
Sensor Size y (mm)	h_s	32.9
Focal Length (mm)	f_{mm}	49.1
Image height (pixels)	H	8736
Image width (pixels)	W	11648

Table 2.1: Camera parameters used for calibration.

The camera model and its sensor size being fixed. The focal length being that of the camera, further altered by its objective. The focal length was automatically measured by the camera at the moment of capture. To fill in the unknowns of our K matrix we have to calculate the focal lengths in pixels. This is achieved in the equation:

$$\frac{W}{w_s} f_{mm}$$

And a corresponding one for height. The first term determines the pixels per millimeter along the axis, then multiplied by the focal length in millimeters. Thus, we have a fully populated K matrix:

$$K = \begin{bmatrix} \frac{W}{w_s} f_{mm} & 0 & c_x \\ 0 & \frac{H}{h_s} f_{mm} & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Camera intrinsics are therefore handled. This is only part of the calibration, the other part of M, the extrinsic matrix is yet to be determined. The extrinsic matrix is what moves points from world space into camera space. The camera's relative position to the world origin, denoted t for the translation vector, and the cameras rotation matrix, R.

$$M = K[R \mid t]$$

R is calibrated using a perspective-n-point (PnP) solver and the known K matrix. Since the church has been mapped using LIDAR to preserve the spatial context for the preservation project, the measurements were easily acquired through their research portal[14]. A world origin was selected and then points pictured on the target wall and the adjacent one were measured for their relative world coordinates, and their matching image coordinates were found on the image. OpenCV's PnP solver with the iterative Levenberg-Marquardt Method was used, in this case with $n = 7$ for each corner point and 3 extra to mitigate errors due to all points on the wall being coplanar, which can be a problem with this method of PnP. There is further discussion on the selection of PnP method in the reproducibility section. The output from this is the cameras translation and rotation vectors. The rotation vector required transformation into the matrix form by the rodrigues rotation formula, also already existing in OpenCV.

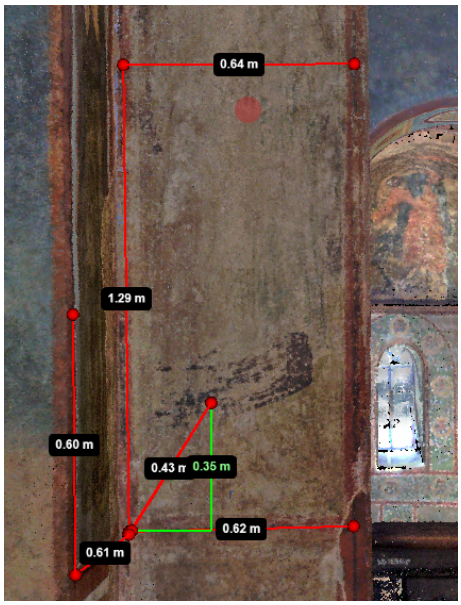


Figure 1: Measurements on the wall, screenshot from GRIDHs portal

2.3.2 Positioning the sphere

With the camera fully calibrated, we can use its intrinsic matrix to calculate the indicator sphere's position in camera space, using its size and position in the image. The center of the sphere in the image was selected manually. Although, algorithmic detection was attempted, manual selection proved more accurate. Since the sphere remains stationary between images, this process only needed to be performed once.

Because the sphere appears dark in all regions except for the specular highlight, its outline is unclear in individual images. To make it easier to locate the sphere's center, an image was created by taking the brightest pixel at each location across all photos in the set. The outline of the sphere was then identified in this compounded image, and its center coordinate was chosen as the center of the circular outline.

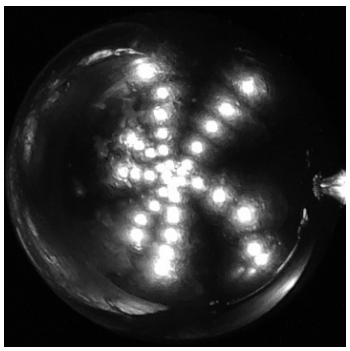
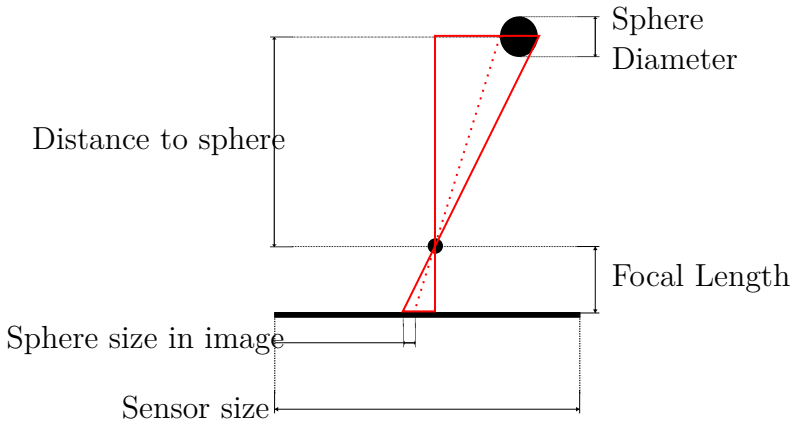


Figure 2: Compounded maximal illumination.

A derivation from the pinhole camera model is then needed to relate the apparent size of an object in pixels on the image to its depth from the camera.



The red triangles are congruent, and are hence the same shape up to a proportional constant a. This constant can be calculated as we know the sphere's size in pixels on the image, s_{px}, and what physical length a pixel corresponds to on the sensor, the sphere projection's width on the sensor is then proportional by a to the sphere diameter, denoted s_d:

$$a = \frac{s_d}{s_{px}\frac{w_s}{W}} = \frac{s_d}{s_{px}}\frac{W}{w_s}$$

This constant is then used to calculate the depth of the sphere, denoted s_z, using the fact that the depth is congruent to the focal length:

$$s_z = af$$

The other components of the sphere's position, s_x and s_y are then calculated using the sphere's image center (u, v) and converting it from a pixel in coordinate space to a ray in camera space and multiply it with the pixel's depth. Again u_x, u_y, denotes the principal points.[15]

$$\begin{vmatrix} u \\ v \end{vmatrix} = \begin{vmatrix} fs_x/s_z + c_x \\ fs_y/s_z + u_y \end{vmatrix}$$

Rearranged to solve for s_x, s_y:

$$s_x = \frac{u - c_x}{f}s_z \qquad\qquad s_y = \frac{v - c_y}{f}s_z$$

Which fully positions our sphere in camera centered coordinates. An intuitive explanation of this equation is that $u - c_x$ becomes a measure in pixels for how far towards the edge an object is from the vertical center of the image. Then this is divided by f, also in pixels, to become a unitless measure of the x-directions ratio to the z direction along that pixels corresponding homomorphic coordinates. Then the multiplication with c_z along z takes the fraction into a dimensioned value along the camera centered x-axis. The transition from camera to world coordinates is not needed, as the later photometric stereo algorithm uses the depth and light positions in camera-centered world space.

To sanity check the results from this section, a scene was set up in blender, containing two planes matching the dimensions of the pictured walls, and a sphere at the calculated world position. A digital camera in the scene was then configured to match the K matrix, and moved to the location specified by the R and t vectors. The camera was then overlaid with an image from the photo set to clearly display the alignment and any possible mismatch between the calculations and the real world.

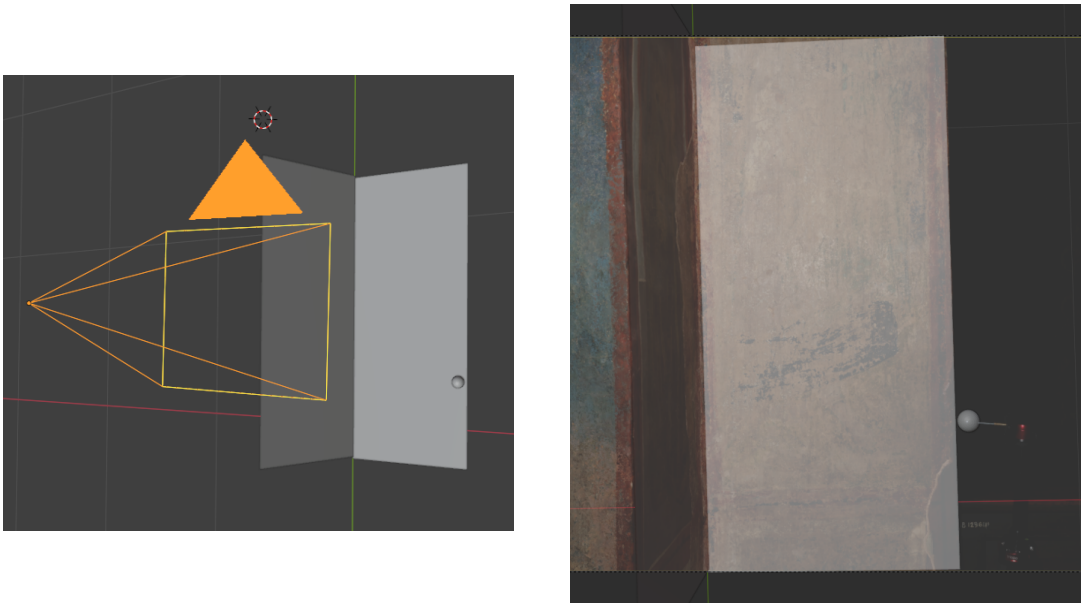


Figure 3: Picture of simulated scene, comparison to actual image

2.3.3 Positioning the light sources

With the sphere in position, the final step before optimization is positioning the light sources. This became a larger part of the project than anticipated, as there is only one indicator sphere. The standard for near-light photometric stereo on real images is either a pre-calibrated lighting array or calibrating light positions as a minimization problem along the indicated directions from two or more spheres.

A combination of techniques was therefore employed. First a ray on which the light lies was calculated using the specular position on the calibration sphere. This ray was then intersected with a plane perpendicular to the wall, positioned along the main light gradient.

2.3.4 Calculating the light source ray

The specular point was then identified by applying a gaussian blur over the image and selecting the brightest point with openCV. The direction from the specular point to the camera was calculated by projecting the specular points image coordinates using the cameras intrinsic matrix in a manner similar to that used when positioning the sphere.

Since the diameter of the sphere is known, a theoretically defined sphere with this

diameter was placed at its calculated real world position. The projected ray to the specular point could then be intersected with the theoretical sphere, positioning the specular point in camera space. The light direction from the specular point is simply the reflection of the vector from camera to specular point around the sphere's surface normal at this point. The normal was calculated by taking the difference from the sphere's center and the specular point position, and the viewing direction is then reflected around this normal.

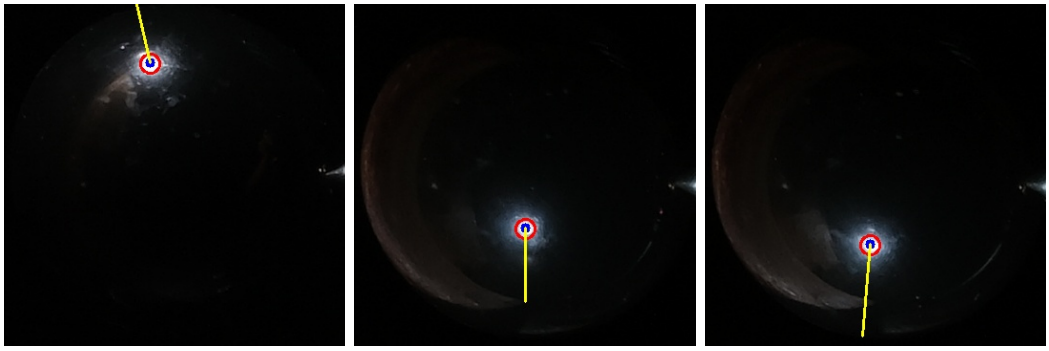


Figure 4: Algorithmically identified specular point and calculated light direction

2.3.5 Calculating the light source plane

The theory behind this step is that, due to the lambertian image model which we have assumed, the square falloff of light intensity will result in the intensities across the wall increasing toward the light source. To quantify this falloff each image was divided by the average light intensity at each pixel, and the ratio image was separated into LAB-color space, where the L stands for light intensity. The intensity channel was isolated and blurred with gaussian blur to reduce the importance of irregularly lit regions due to the unevenness in the wall. From this blurred image, the sobel gradient was calculated for each pixel using a 5x5 kernel[16]. An approximate light direction was calculated by summing the gradients of all pixels. The plane was then placed intersecting the brightest point on the wall and spanned by the wall normal and the calculated gradient.

Figure 5: Calculated light plane on the blurred ratio image, the red line is the gradient direction, with the end point in the brightest pixel on the wall.

2.3.6 Final light source estimation

The light source is then positioned along the ray calculated from the sphere, at the intersection with the plane calculated from the light gradient. The intersection point is then used as the light source position. This is done for each image in the photo set, and saved in a file for later use in the optimization step.

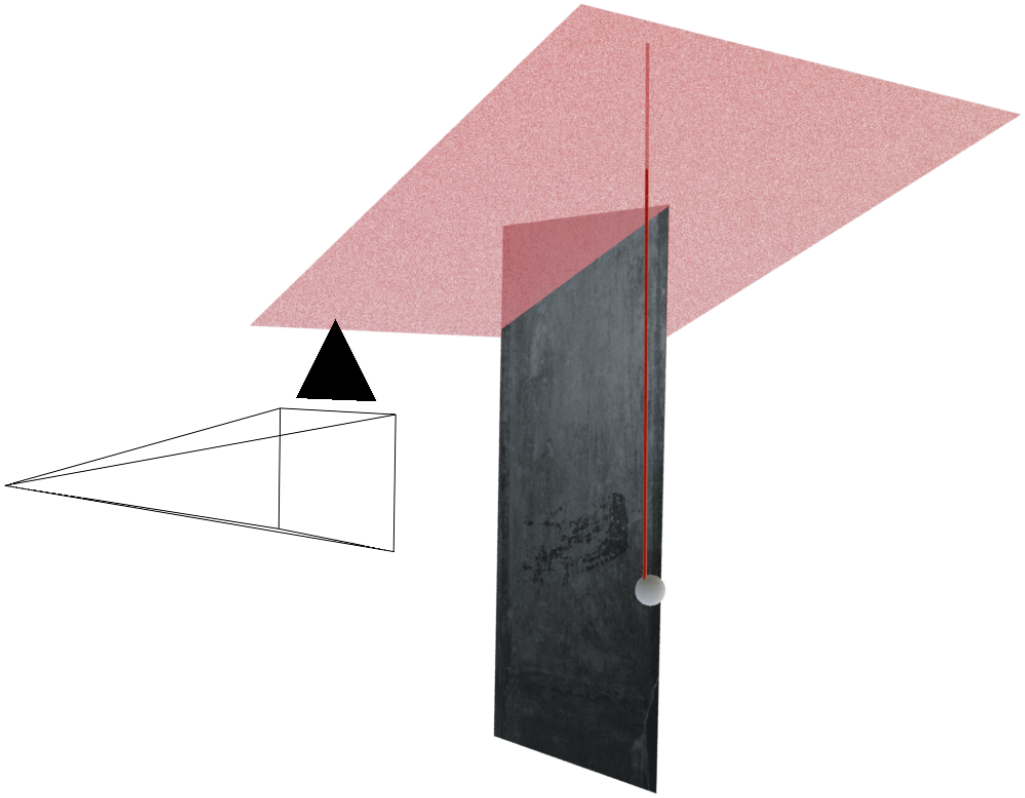


Figure 6: Intersection of the ray and the plane, the red line is the ray from the sphere, the red plane is the light plane.

While the direction from the sphere is of high certainty, the more heuristic light plane estimation introduces errors due to the intensity not only varying according to the falloff. Sometimes the strongest gradients point towards regions with specular highlights or regions that are brighter due to uneveness in the wall. While using more than one layer of neighbours when calculating the sobel gradient along with gaussian blurring mitigates these factors and largely eliminates them on a pixel level, larger regions of unevenness or specularity still skew the estimated light direction. These errors are later taken into account through error handling in the optimization model.

2.4 Depth map creation

What remains of the implementation is then the creation of the depth map. Many different problem definitions and strategies for highlighting inscriptions were tried while the photometric model at large remained the same. More on the variations in the result section.

The solver used in the implementation was the ceres solver developed by Google. This is due to its versatility in defining loss functions and also ceres ability to automatically differentiate them numerically. This is of importance due to the jacobian computation otherwise having to be defined in the program, greatly increasing the difficulty of prototyping. It being implemented in C++ is also of importance, other libraries in python were tried but handling many images of a large size proved challenging in python but worked well in C++, along with other aspects such as easier parallelism when reading images.

The optimization problem largely follows the modeling in the paper from Quéau et al.[3] The expected light is calculated according to the lambertian light model as a function of surface normal, distance to light, albedo and light intensity. The error compared to the observed light intensities in the photo set computed and minimized for.

Minimization then happens in two alternating steps. After initialization a first depth map is calculated, and then this depth map is used to refine the albedo estimates. As discussed in the introduction, other penalties were also introduced to possibly highlight engravings.

2.4.1 Data Formatting

To format the data in a way accessible to the solving algorithm, the images were processed into an observation matrix I, where each measured pixel intensity $I_{j,i}$ is stored in row j corresponding to pixel j of column i, corresponding to image i. Since images are 2-dimensional, they had to be collapsed into a 1-dimensional vector, which was done by stacking each column from left to right, so called column major order.

To isolate the light intensity of each pixel, the images were again converted into LAB-color space and the L-channel isolated. After the stacking of the images into the observation matrix I a heuristic ρ vector was initialized by taking the row wise average of I. The theory behind this step is that under lighting conditions of varied direction but similar intensity, the average reflected intensity should be similar to the albedo of the surface,[17] a good starting point for the optimization.

The z vector was initialized to the depth of the camera calculated by the camera calibration. The z vector is also to be viewed as an image in column major order. Then a struct with all constant values was created, such as the K matrix, its inverse and inverse transpose. All of these were done as a one off during initialization, to avoid doing it in the computationally intense optimization loop.

2.4.2 Calculation of the photometric residual

First the partial derivative of the heightmap is calculated at a pixel $j_{u,v}$, using finite forward differences:

$$\nabla z_{u,v} = \begin{vmatrix} z_{u+1,v} - z_{u,v} \\ z_{u,v+1} - z_{u,v} \end{vmatrix}$$

Central difference was also tried, and worked better to accentuate large engravings while overly smoothing small engravings and taking longer to calculate.

To transform this into a surface normal used in lighting calculation it has to move from a difference in pixel space into a camera centered world normal. To do this transformation the inverse of the K matrix is used:

$$n = K^{-T} \begin{vmatrix} \nabla z_{u,v} \\ -1 \end{vmatrix}$$

The calculations inside the automatic differentiator happen hundreds of thousand of times, even if only a small region of the wall is optimized over, so any saved calculations have a large impact. This plays a large role in the following equations explained below.

A geometric term s is calculated, accounting for light direction and falloff, the vector from light source i to pixel j denoted $l_{j,i}$

$$s_{j,i} = \frac{l_{j,i}}{|l_{j,i}|^3}$$

The denominator being cubed is accounted for by one power to normalize the light direction, and two powers to scale it by the square falloff with distance. The incident light term ψ is then the cosine term of the incident light and the normal.

$$\psi_{j,i} = s_{j,i}^T (K^{-T} \begin{vmatrix} \nabla z_j, i \\ -1 \end{vmatrix})$$

And the residual $r_{j,i}$ of the optimization is then the difference between the measured light level of pixel j in image i, denoted $I_{j,i}$, and the incident lighting scaled by the albedo ρ_j

$$r_{j,i} = I_{j,i} - \rho_{j,i}\psi_{j,i}$$

An important note here is that the surface normal is never normalized, so a steep surface change will have a normal of a larger magnitude. The result of this is that normals and albedos are coupled. This is of course not true in the strict physical sense as the incident lighting varies only with surface normal. However, if this is not the case there is a linear ambiguity between albedo and normal, as the pixel density varies across the image and becomes sparser towards the edges. Hence the same surface normal direction can be computed with different variations in height

for different pixels in the image, but if it is coupled to the magnitude this effect is mitigated. So the real world albedo $\hat{\rho}$ is actually:

$$\hat{\rho} = \frac{\rho}{|K^{-1}\begin{vmatrix}\nabla z_j, i\\ -1\end{vmatrix}|}$$

This allows the solver to find a unique minimizing solution for any given pixel, and it not varying ambiguously with perspective. Note that the light was assumed to be constant and non anisotropic, with the albedos themselves compensating for light intensity. This is only possible since all images used the same LED for lighting the scene, and the LED is either not anisotropic or directed in a manner such that anisotropy does not vary across the wall.

In addition to the photometric residual, a smoothness term was experimented with, in part to aid in convergence and in part to reduce high frequency noise. This proved efficient for convergence, but at cost of detail. The smoothing during optimization was also hard to tell apart from smoothing done of the produced mesh afterwards, so it was not used in the final implementation. The smoothing used was laplacian smoothing:

$$r_{u,v} = z_{u,v} - \frac{1}{4}(z_{u+1,v} + z_{u-1,v} + z_{u,v+1} + z_{u,v-1})$$

The pixel depth, subtracted by the average depth of its neighbours. Laplacian smoothing was chosen because it has a negative value for protrusions and a positive value for indentations. This means that a simple *max* operation can be used to penalize protrusions only, while not penalizing engravings.

2.4.3 Solver Setup

Ceres was set up in accordance with the recommended settings for a large variable sparsely connected problem in the Ceres documentation [18]. This problem is referred to as sparse, since there are many thousands of variables and residuals, but each residual is only tied to 3 variables, the corresponding depth and its two forward neighbours. The solving had to be done in patches of the wall, with regions around 900x900 pixels taking in the vicinity of 20 iterations to converge, each iteration taking about 30 seconds, but the depth map starts receiving diminishing improvements after about 4 depth and 3 albedo iterations. Even with smaller regions of interest, the sparsity of the problem had to be used or Ceres quickly used all available memory and could not continue.

Thus the Sparse Normal Cholesky solver was used, combined with Jacobian preconditioning. Jacobian preconditioning was used because it better sets the problem up for sparsity and parallelism, optimizing interlinked variables at the same time to avoid cache misses.

For the depth step, a Cauchy loss function was used, although Tukey loss also produced good results. Cauchy and Tukey loss have in common that they heavily reject outliers, which is important due to the error in lighting positions and the

approximative lighting model, so they will not factor in datapoints such as specular highlights, self shadowed pixels, or incorrect light positions. The ideal loss function and delta parameter was not conclusively decided, as different values accentuated different features, but the one that was proved to work for most cases was cauchy with a delta of 0.1. If the delta is lowered more high frequency noise is included in the depth map, but smaller inscriptions are more apparent than for a large delta where they are overly smoothed over.

For the albedo step, cauchy loss with a larger delta of 0.5 was used, but the difference in results between deltas was not as pronounced.

A callback function running after each iteration to save the depth map and albedo values was implemented for both steps of the optimization, saving the current depth map as an image and a .obj file containing the mesh. This greatly aided in debugging as it allowed the examination of how fast the depth map converged and how the albedo values changed. The callback function also allowed for the optimization to be interrupted at any time, which was useful when testing different configurations of the solver.

when trying to calculate the depth map of the entire wall, the calculation had to be split up in sections. For this a region of interest strategy was devised, where chunks of the wall were optimized separately. The region of interest has modular area and modular overlap, and after complete operation the regions were stitched together by taking the average of the overlapping pixels. This also allows for regions of interest to be selected manually and optimized for as a list, which was useful when testing which configuration produced the best results for different engravings.

2.4.4 Algorithm overview

The following algorithm summarizes the main steps involved in the creation of the depth map from the input images. It consists of the initialization, data preparation, light position estimation, and the iterative optimization process used to produce the depth maps for each region of interest and subsequently the whole wall:

Algorithm 1 Depth map creation

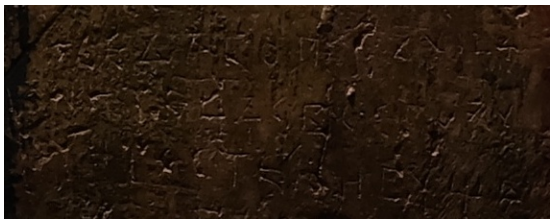
```
Choose one image for coordinate initialization
Choose a world origin
Define points on the wall in 3D space
Input K matrix from camera calibration
Solve for camera position using K matrix and world points
Position reflectance sphere(s)
for each image do
    Read image
    Convert to LAB color space
    Isolate L-channel
    Calculate light plane          ▷ Only one sphere was available, non standard
    Intersect light plane with light direction from sphere
    Save light position
for each image do
    Read image
    Convert to LAB color space
    Isolate L-channel
    Convert image to column major vector
    Stack image into observation matrix I
initialize regions of interest
for each region of interest do
    Initialize albedo ρ vector
    Initialize depth z vector
    while not converged or max iterations do
        Run depth optimization
        Save depth map
        Run albedo optimization
        Save albedo map
if Stitching then
    Stitch regions of interest
    Save stitched depth map
```

3

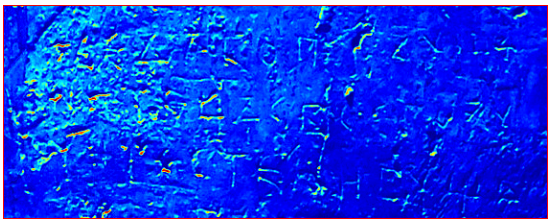
Results

The result section will go over examples from the images compared to a photo image of the same region. The picture was taken with raking light, shining from a steep angle to accentuate engravings. All examples also include evaluation with TVT, followed by a discussion on performance. The examples have been selected from different parts of the wall, and contain inscriptions of varying depth and clarity, allowing the evaluation of when an inscription can be missed by this algorithm.

3.0.1 Small Letters Example



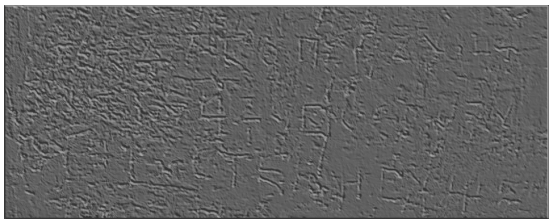
(a) Raking light picture



(b) Depth map

(c) TVT enhanced topography map

(d) Z-axis visualization



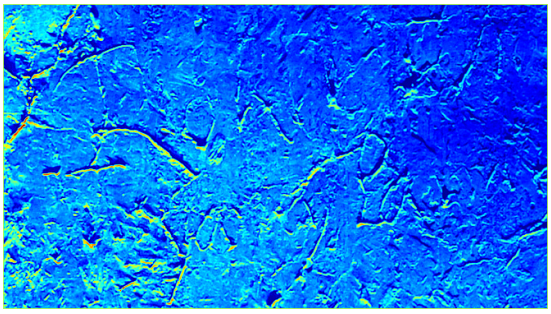
(e) Artificial relighting

Figure 1: Visualizations of the Small Letters example

An example with small letters, they are quite shallow, and while they usually appear in raking light, their readability as a whole is usually poor. This example clearly highlights the value in the visualization of the mesh. Note here especially that TVT manages to enhance letters even in the roughly textured region in the top left corner, which is hard with raking light as the rapid alternating light and shadow makes inscriptions hard to spot.

3.0.2 R Example

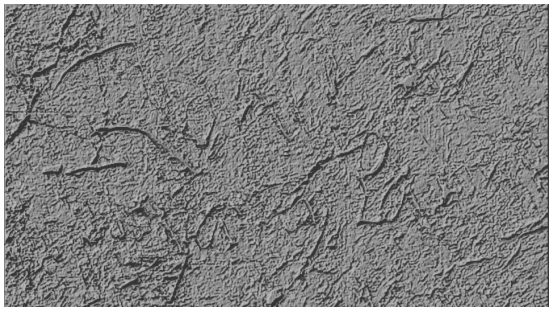
(a) Raking light picture



(b) Depth map

(c) TVT enhanced topography map

(d) Z-axis visualization



(e) Artificial relighting

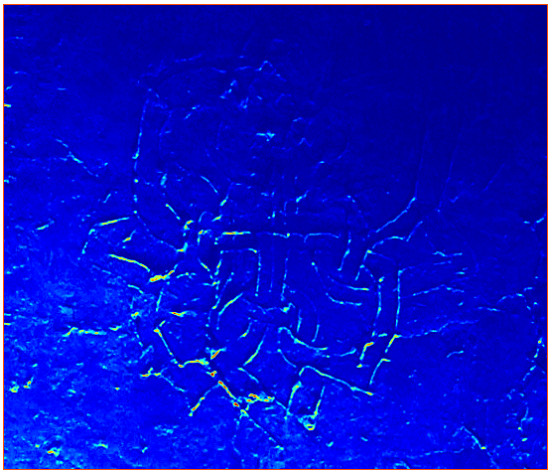
Figure 2: Visualizations of the R example

This example was chosen because while much of the inscription is deep and usually obvious in raking light, the curving sections of the letters are shallower. Such as the top bit of the δ next to the R. In this case, the bowl of the letter is deep while the ascender is very shallow and only a few pixels across, but it is intelligible in the

visualization. While it would be hard to spot without the visual cue from the bottom part of the letter it increases readability of the letters as a whole.

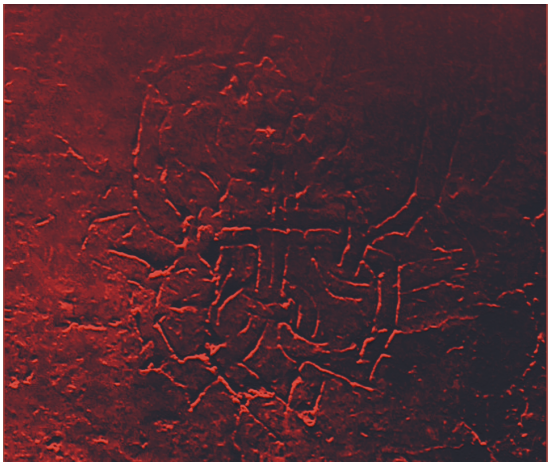
3.0.3 Cross Example

(a) Raking light picture

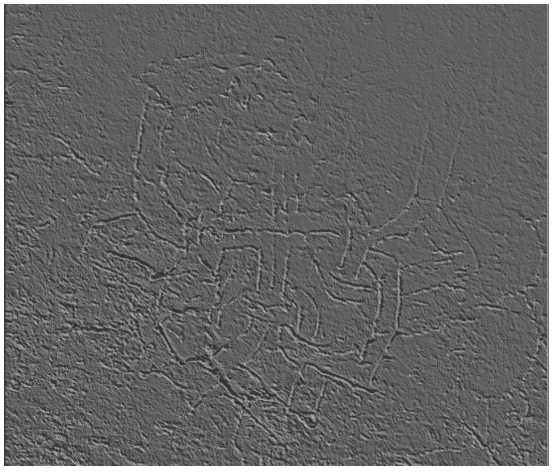


(b) Depth map

(c) TVT enhanced topography map



(d) Z-axis visualization

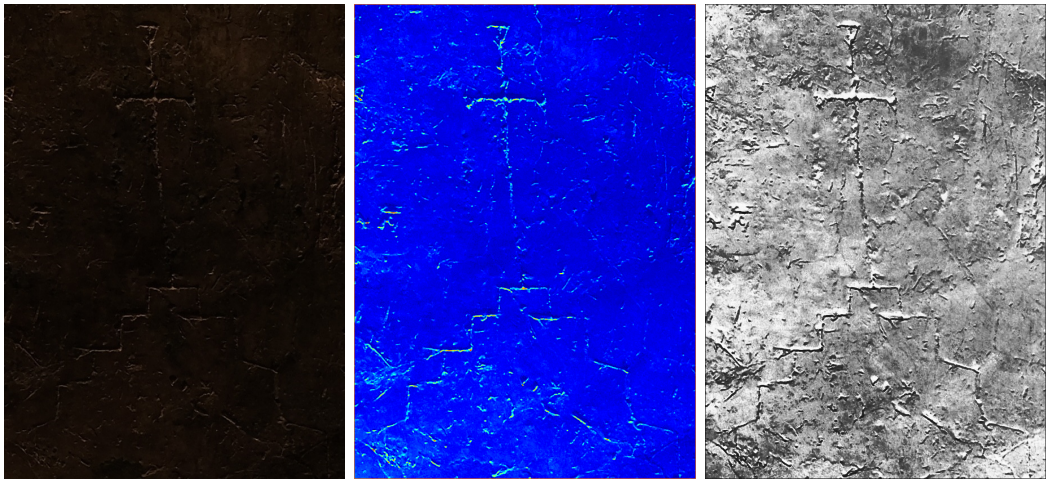


(e) Artificial relighting

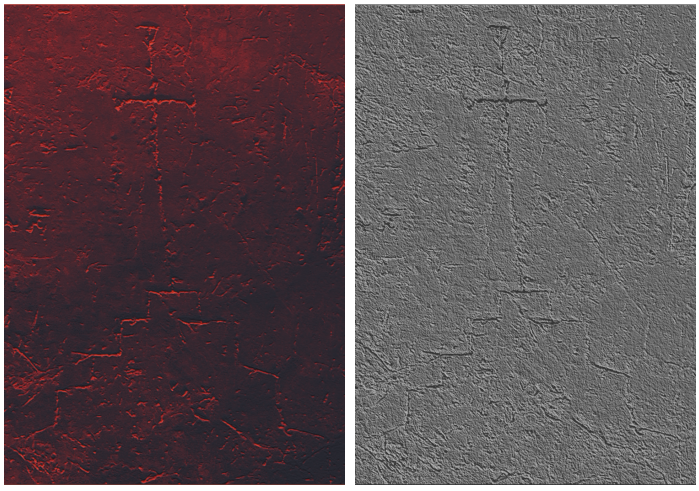
Figure 3: Visualizations of the Cross example

The varying depth in this image is a good example of when the algorithms limit is reached, as a large part of the cross is readable but the top right corner is shallow and thin and disappears almost completely in the mesh. Like in the example above however, with the contextual clues given from the rest of the cross it is slightly visible.

3.0.4 Sword Example



(a) Raking light picture (b) Depth map (c) TVT enhanced topography map



(d) Z-axis visualization (e) Artificial relighting

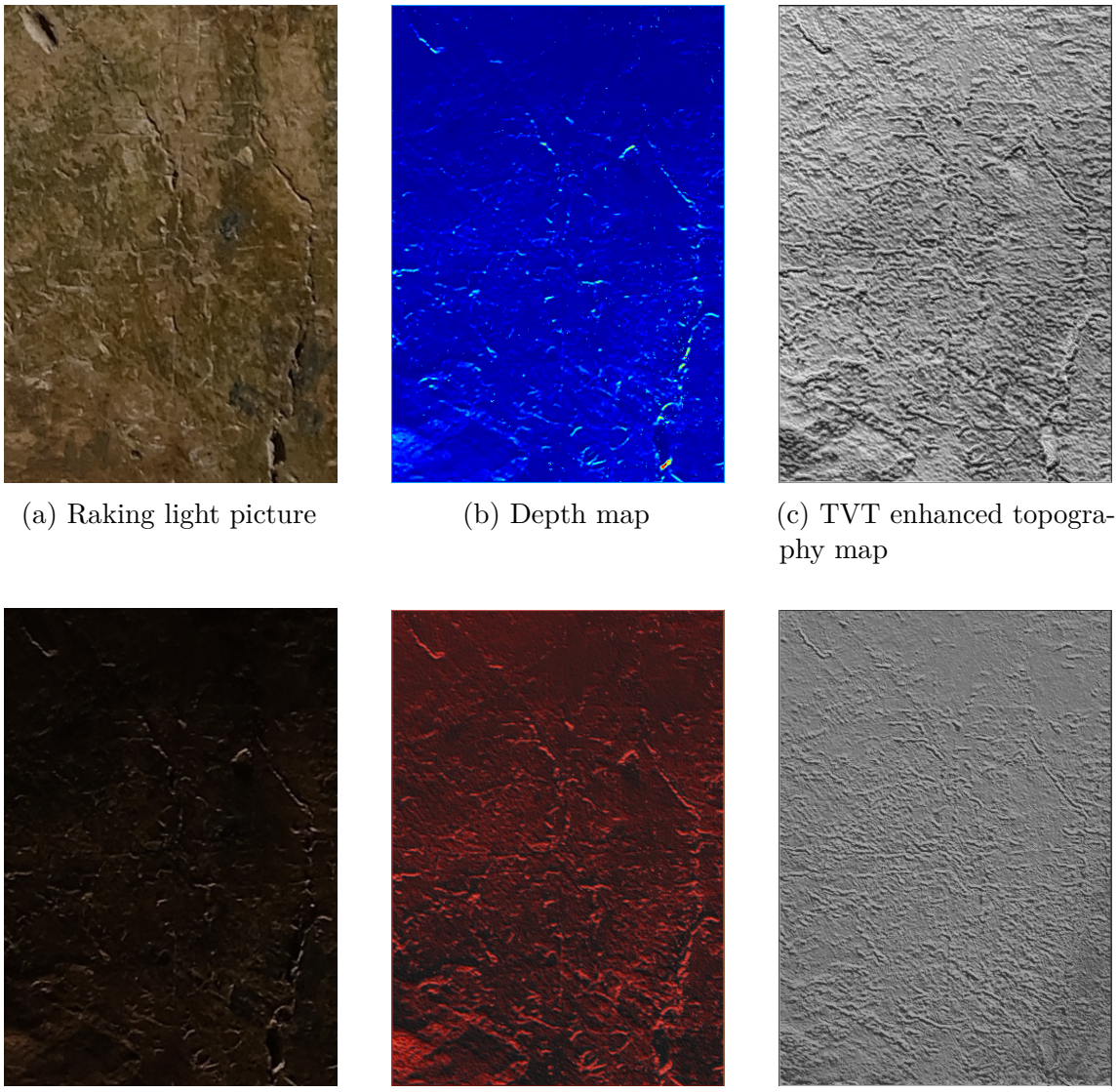
Figure 4: Visualizations of the Sword example

This example was chosen due to how well TVT isolates the inscription, even though it is quite thin throughout.

All of the above are examples of the algorithm working well, but there are also cases where it fails. The algorithm is not able to preserve engravings that are too shallow, for example this double barred cross. The cross appears quite prominently using

raking light, but is difficult to spot in the mesh, with all of the techniques used above:

3.0.5 Double Barred Cross Example



(a) Raking light picture

(b) Depth map

(c) TVT enhanced topography map

(d) Steeper raking light picture

(e) Z-axis visualization

(f) Artificial relighting

Figure 5: Visualizations of the Double Cross example

This is an example of an inscription that is missed by the algorithm. It is shallow and thin throughout. Though, the appearance in the photos might also be due to the base material of the wall being a different colour to the that of the paint which was scraped off while inscribing, since the cross is very faint on images with a steeper angle of raking light.

As you can see from the above examples, the algorithm preserves both obvious and

some subtle engravings. Although it performs worse than specified RTI equipment, photometric stereo could be a usable tool while remaining cost effective and portable.

The value of the mesh compared to a raking light picture or a normal map is hard to communicate over text, as the added dimension is hard to convey. Scaling and shading along the z axis allows for fine tuning of level of detail and isolation of different engravings at different depths. Different engravings can be isolated and marked down iteratively until a clearer whole takes shape. This shading along the z-component is what the panels above try to visualize but it is best experienced interactively.

3.1 Performance

The algorithm performs well in the sense that it converges, while the time it takes to converge is not ideal. One solver iteration for both depth and albedo on a 900x900 pixel region takes around 30 seconds on a 16 threaded CPU with 16 gigabytes of RAM. GPU-acceleration was tried and while it did improve performance marginally for smaller regions of interest it could not process 900x900 patches, so using CPU computation only was more performant as it reduced the number of patches required in total.

While convergence takes around 10-15 iterations, diminishing results to the point where two iterations are visually indistinguishable, usually takes around 3-4 iterations. The entire wall was ran in 900x900 patches for 5 iterations and it was finished in 2 hours and 38 minutes. While it is slow, the computation time is not a bottleneck in the aspect of photometric stereo being more budget efficient than RTI, since the PC it was ran on is consumer grade.

3.2 Discussion and Future Work

There are a number of areas for further research that could possibly aid in preserving even more of the depth data using photometric stereo. Aside from making this algorithm into a user friendly program, the main areas of improvement are noise reduction and suppression. When it actually comes to the optimization, the data capture and calibration steps have introduced noise, the reduction is about reducing the calibration error, while the suppression is about handling the noise in the optimization.

3.2.1 Noise Reduction

There were many noise reduction methods already in place, such as having a fixed exposure to make lighting values consistent between images, and having a camera measured focal length. However the PnP problem could be further refined to get better camera extrinsics, which are at the fundamental step as the sequential sphere, light and pixel positioning all use this data.

Specific to inscriptions, a PnP algorithm using geometric primitives could be used. This could be helpful since triangles or rectangles are easier to measure on a wall in reality. Other real world applications of a similar algorithm would likely not have such a useful tool as the saint sophia database where measurements are easily taken in 3-dimension. The PnP solution in this algorithm is also reliant on the neighbouring wall, as the PnP solver used does not support coplanar points only. This is a limiting aspect since other inscriptions might not have a neighbouring wall or other objects that are easy to position relatively to the wall. Thus a PnP algorithm that is specified on coplanar objects such as a wall could probably reduce noise. This is a research area in its own and has been researched in the past, such as in this paper by Oberkampf et al [19], and it could possibly aid in the preservation of inscriptions.

Another approach could be to not position the wall at all. Solely placing the origin at the camera and multiple indicator spheres in camera coordinates, and then the light sources directly in camera space. The depth map could then be initialized to a flat plane at a single measured distance from the camera. This approach would likely take longer to converge, as the initial depths could be further away from the actual but it would be less reliant on real world measurements. This approach could not be used in this project due to the single indicator sphere requiring the placement of the wall for light source calculation, but it could prove more usable in reality than measuring points for placement using PnP.

3.2.2 Noise Suppression

The noise suppression played a large role in this algorithm, but it is still a fixed loss function from the Ceres library. The loss function could possibly be improved by adding some layers of adaptivity to it, such as scaling the delta value based on the calculated residuals. A progressively evolving loss function could also be tried, where

initially the residual suppression is very low to make use of all possible data, then progressively increasing the suppression to make the best possible fit to the well calibrated light sources. This could be used on a per pixel basis, to discard erroneous data due to shadows or specular highlights, and it could be combined with a light mask to completely discard ill placed light sources.

This photo set does not have many specular highlights, but in the interest of generalizing to other walls, a method to detect specular highlights could be used. This could be done by using a specular highlight mask, and subsequently discarding pixels deemed to be specular. The chapter by Liu et.al on specular highlight in the book [20] describes multiple methods for this. An initial starting point could be to compute specular masks using a threshold based on the estimated albedo of the pixel, and masking it if it appears in the color of the light source, as specular highlights are not colored by the material. To preserve more data points further techniques to remove specular highlights could also be tried.

3.2.3 First guess improvement

The first depth guess could be improved with a more sophisticated method than a flat plane. For example by capturing a couple of calibration images from different camera angles, all containing the same calibration spheres, which would give the images approximate relative camera coordinates. This could give an initial shape estimate of the wall using shape from parallax methods, and then refine this shape on a more detailed level using photometric stereo optimization.

3.2.4 Light Source Positioning

Light source refinement could be added as another alternating step in the optimization. This was not done in this project, as the smaller regions of interest left a lot of ambiguity in the optimization of the light source positions and not having them fixed diminished the quality of the results.

If the wall as a whole was subsampled, the intensity difference across the wall could help in positioning the light sources more accurately, before smaller regions of interest are optimized in full resolution. This could also possibly be paired with a manual initial guesses of light source positions, to remove the need for a reflective sphere, further reducing the required hardware.

3.2.5 Automatic Positioning

To truly make this method of photometric stereo available to the general public, further reducing the required hardware list to even more widely available devices such as phones could be an area of research. The positioning could be done using stickers placed around the area of interest. The stickers could contain QR-codes of known measurements for easy algorithmic detection in the photos, QR-code detection being available in OpenCV. Then one phone could be fixed on a tripod with the flash disabled, and one phone could be set to use flash and be moved around while

capturing photos simultaneously. Then the QR-code positions can be solved for in the stationary device's camera coordinates using the known dimensions, similarly to the sphere in this paper. The QR-positions could then be used to solve a PnP problem from the perspective of the second device, positioning its camera and light source in the stationary device's camera coordinates.

This could reduce required hardware to only two mobile phones, a tripod, and a printer. This would likely work for any two phones that can capture RAW image data, for which the exposure can be fixed, and the anisotropy of the flash could be either calibrated or diffused using a diffuser.

This would however reduce the amount of data captured at once, as mobile phones have a lower resolution than the DSLR camera used in this project. This could be mitigated by splitting the wall into sections based on the resolution of the camera in relation to the size of the smallest inscriptions, placing the phones at a distance where the smallest inscriptions are at least a few pixels across. Another possibility is having the stationary device remain as a dedicated high resolution camera while the device used for flash is a phone.

GRIDH have tried the LiDAR capabilities of mobile phones in scanning rock carvings and other inscriptions. While they make a good and well proportioned 3D model of entire objects, inscriptions are usually too shallow to be captured by the LiDAR sensor, hence photometric stereo could possibly preserve engravings better.

3.3 Ethical Considerations

Since this is a conservation effort in an active warzone there have been ethical questions at GRIDH concerning the collection of data and its distribution. The risks taken during data collection is not considered in this project as the images captured for this evaluation are part of GRIDHs larger project with its own risk analysis.[21]

What this project could provide to the larger, or future projects is however reduced time and budget to capture 3D-models of a wall. This could mitigate the risks taken by reducing time spent in a vulnerable area. A less costly technique, if applied to other walls, could also make digital conservation available to parties previously hindered by costs.

Another important aspect that GRIDH has considered for cultural importance is preserving walls in their entirety. While preserving more prominent or culturally important inscriptions in themselves is cheaper and less costly, preserving the entire wall serves purposes in itself. It could make the perceived importance of inscriptions at the time of writing more apparent, if they are central and large. Further, it could include data otherwise not deemed worthy of capturing, such as small illicit graffiti, which could in itself potentially have meaning in other research projects. Such as mapping which groups have had access to which areas of the church or mapping travel patterns of visitors through the ages.

Preserving the whole of the cultural value is especially important in conflict zones, as destruction by warfare is largely random and could affect any part of the church. It is also important to deter and counteract any purposeful destruction that aims to erase specific cultural traces, by for example removing or altering graffiti attributed to certain demographic groups while preserving others. This can be done in a pattern that aims to show history from a perspective that furthers a specific agenda. This is a tactic widely used in territorial wars where the offender wants to impose a cultural possession over a targeted area[22].

3.4 Conclusion

Photometric stereo has been shown to be a usable tool for the preservation and analysis of inscriptions. While the initial conditions for this project were not ideal, the algorithm was able to produce a mesh that maintained readablity of even subtle engravings, while the most shallow and thin are at best faintly represented in the mesh. Further research could be done to improve the algorithm, such as improving the noise suppression and reduction, and making the algorithm more user friendly. The algorithm is not a replacement for RTI, but it can be used for the preservation of inscriptions in a cost effective manner.

Bibliography

[1] University of Gothenburg, *DIGICURE: Digital Cultural Resilience and Protection*, Accessed: 2025-06-19, 2025. [Online]. Available: https://www.gu.se/forskning/digicure-digital-cultural-resilience-and-protection.

[2] C. Horn, O. Ivarsson, C. Lindhé, R. Potter, A. Green, and J. Ling, "Artificial intelligence, 3d documentation, and rock art—approaching and reflecting on the automation of identification and classification of rock art images," *Journal of Archaeological Method and Theory*, vol. 29, no. 1, pp. 188–213, 2022.

[3] Y. Quéau, T. Wu, and D. Cremers, "Semi-calibrated Near-Light Photometric Stereo," en, in *Scale Space and Variational Methods in Computer Vision*, F. Lauze, Y. Dong, and A. B. Dahl, Eds., vol. 10302, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2017, pp. 656–668, ISBN: 978-3-319-58770-7 978-3-319-58771-4. DOI: 10.1007/978-3-319-58771-4_52. [Online]. Available: http://link.springer.com/10.1007/978-3-319-58771-4_52 (visited on 03/26/2025).

[4] T. Papadhimitri and P. Favaro, "Uncalibrated Near-Light Photometric Stereo," en, in *Proceedings of the British Machine Vision Conference 2014*, Nottingham: British Machine Vision Association, 2014, pp. 128.1–128.12, ISBN: 978-1-901725-52-0. DOI: 10.5244/C.28.128. [Online]. Available: http://www.bmva.org/bmvc/2014/papers/paper127/index.html (visited on 05/13/2025).

[5] R. J. Woodham, "Analysing images of curved surfaces," *Artificial Intelligence*, vol. 17, no. 1, pp. 117–140, 1981, ISSN: 0004-3702. DOI: https://doi.org/10.1016/0004-3702(81)90022-9. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0004370281900229.

[6] B. K. Horn and R. W. Sjoberg, "Calculating the reflectance map," *Applied optics*, vol. 18, no. 11, pp. 1770–1779, 1979.

[7] A. Hertzmann and S. Seitz, "Example-based photometric stereo: Shape reconstruction with general, varying brdfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1254–1264, 2005. DOI: 10.1109/TPAMI.2005.158.

[8] M. McGuigan and J. Christmas, "Automating rti: Automatic light direction detection and correcting non-uniform lighting for more accurate surface normals," *Computer Vision and Image Understanding*, vol. 192, p. 102 880, 2020, ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2019.102880. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314218302686.

[9] D.-Ø. E. Solem and E. Nau, "Two new ways of documenting miniature incisions using a combination of image-based modelling and reflectance transformation imaging," *Remote Sensing*, vol. 12, no. 10, 2020, ISSN: 2072-4292. DOI: 10.3390/rs12101626. [Online]. Available: https://www.mdpi.com/2072-4292/12/10/1626.

[10] Z. Zhang, "A flexible new technique for camera calibration," en, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000, ISSN: 01628828. DOI: 10.1109/34.888718. [Online]. Available: http://ieeexplore.ieee.org/document/888718/ (visited on 01/31/2025).

[11] C. Ware, "The Environment, Optics, Resolution, and the Display," en, in *Information Visualization*, Elsevier, 2013, pp. 31–68, ISBN: 978-0-12-381464-7. DOI: 10.1016/B978-0-12-381464-7.00002-8. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/B9780123814647000028 (visited on 05/20/2025).

[12] Ruo Zhang, Ping-Sing Tsai, J. Cryer, and M. Shah, "Shape-from-shading: A survey," en, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 8, pp. 690–706, Aug. 1999, ISSN: 01628828. DOI: 10.1109/34.784284. [Online]. Available: http://ieeexplore.ieee.org/document/784284/ (visited on 11/28/2024).

[13] D. Hutchison, T. Kanade, J. Kittler, *et al.*, "Face Image Relighting using Locally Constrained Global Optimization," en, in *Computer Vision – ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., vol. 6314, Series Title: Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 44–57, ISBN: 978-3-642-15560-4 978-3-642-15561-1. DOI: 10.1007/978-3-642-15561-1_4. [Online]. Available: http://link.springer.com/10.1007/978-3-642-15561-1_4 (visited on 05/13/2025).

[14] University of Gothenburg, *Saint Sophia Research Portal*, Accessed: 2025-03-21, 2025. [Online]. Available: https://saintsophia.dh.gu.se/.

[15] *Opencv documentation*, Itseez, May 2025. [Online]. Available: https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html.

[16] R. Szeliski, "Computer vision: Algorithms and applications, 2nd edition," en,

[17] D. Miyazaki, Y. Onishi, and S. Hiura, "Color photometric stereo using multiband camera constrained by median filter and occluding boundary," *Journal of Imaging*, vol. 5, no. 7, 2019, ISSN: 2313-433X. DOI: 10.3390/jimaging5070064. [Online]. Available: https://www.mdpi.com/2313-433X/5/7/64.

[18] S. Agarwal, K. Mierle, and T. C. S. Team, *Ceres Solver*, version 2.2, Oct. 2023. [Online]. Available: https://github.com/ceres-solver/ceres-solver.

[19] D. Oberkampf, D. F. DeMenthon, and L. S. Davis, "Iterative Pose Estimation Using Coplanar Feature Points," en, *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 495–511, May 1996, ISSN: 10773142. DOI: 10.1006/cviu.1996.0037. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1077314296900375 (visited on 05/22/2025).

[20] Q. Liu, H. Wang, Z. Ma, *et al.*, Eds., *Pattern Recognition and Computer Vision: 6th Chinese Conference, PRCV 2023, Xiamen, China, October 13–15, 2023, Proceedings, Part IX* (Lecture Notes in Computer Science), en. Singapore: Springer Nature Singapore, 2024, vol. 14433, ISBN: 978-981-99-8545-6 978-

981-99-8546-3. DOI: 10.1007/978-981-99-8546-3. [Online]. Available: https://link.springer.com/10.1007/978-981-99-8546-3 (visited on 05/22/2025).

[21] University of Gothenburg, *Digital documentation of inscriptions in the saint sophia cathedral in kyiv*, Accessed: 2025-02-10, 2024. [Online]. Available: https://www.gu.se/en/research/digital-documentation-of-inscriptions-in-the-saint-sophia-cathedral-in-kyiv.

[22] Jönsson, Lars-Eric and Svensson, Birgitta, "Ord på vägen," swe, in *I industrisamhällets slagskugga : om problematiska kulturarv (Dokumentation och forskning från Jämtlands läns museum)*. Carlsson Bokförlag, 2005, 8–24, ISBN: 91-7203-672-9.